REMARKS

The title is amended to reflect the goals of the present invention in a comprehensive manner.

The amendment in lines 1-2 is provided in response to the statutory requirements that the applicant insert a reference to applicant's PPA.

To represent the far reaching effects the present invention will have on future computer architectures, the applicant would like to first provide some important limitations of prior art computers and then present overview of the major features that make the present invention unobvious and hence patentable.

The applicant would like to respectfully point out that this is a basic and new invention. The prior art cited by the examiner has been a mostly improvements of existing architectures machines and the advancement of the art is limited to improvements. The concepts proposed in the present invention advances the state of the present art by solving problems considered unsolvable.

The cache memory based computer system is a probabilistic entity. All probabilistic systems eventually fail in performing its intended task, which is required of the probabilistic systems. The present invention transforms the prior art cache memory based computer system from the probabilistic system to deterministic system.

Cache memory based systems operating as probabilistic system has a finite success ratio. The cache will fail in obtaining required data or instruction during certain operating conditions. Even with a very high hit ratio of 99%, failure can lead to major operational and data integrity problems.

This failure will lead to refill of cache memory that requires a long time interval to achieve the refill. During this time period, serious problems can occur such as execution

delays, loss of input values and a system crash can occur. With the applications program getting larger and more complex requiring broader usage of all resources of a given computer system, a denial of a major resource such as timely availability critical data not located on prior art cache memory can lead to serious failures and problems. These problems become more complicated when the computer system is part of a larger network providing important data to other nodes.

The differences between the present invention and the prior art computer systems are in every area of design, operation and performance.

The cache coherency and consistency problems are eliminated in the new design. Parallel and multiprocessor operation is more reliable. The prior art Multiprocessor and parallel processing systems have serious problems with data integrity and interprocessor communication is limited duel to the problems related to cache coherency and cache consistency. The deterministic nature of the computer system eliminates the need for updating the stale data and have two copies of updated data. Multithreading is accomplished in more reliable and
The strengths of the present invention lie in the deterministic real time operation. The data and instructions can be loaded in real-time and the computer can be started before the entire memory and data is loaded in the computer.

The data and instruction can be transferred in input as well output mode giving the system a real-time duplex operation.

Another major advantage to be realized is in the area of power management. Power management can be made more effective than any thing available in the prior art. The Deterministic nature of fhe computer system also offers advantages to the areas of power usage and power management.

In fact it is possible to develop and query the power management envelope at compile time and it is possible to estimate power dissipation and usage at the compile time which is not possible in prior art systems with cache memories.

New generation processors are facing a serious crisis in terms of power dissipation. The power

The limitations of cache memory based systems has created may hurdles in the development of next generations of processors. These limitations are first described. The many advantages of the present invention has been stated after indicating the limitations of cache memory based computer systems as follows:

LIMITATIONS OF CACHE MEMORY BASED SYSTEMS

1. Processor execution time will always be faster than the memory access time
2. Cache memory is used to fill the gap between the two operating speeds
3. Computer performance is based on cache memory size and speed, which is not a desirable feature
4. Cache memory is a probabilistic system with inherent problems
5. Deterministic system is always superior to probabilistic systems
6. Cache memory miss can create timing problems and create system crashes
7. Power usage and power management is not efficient
8. Power usage is further increased since program execution occurs from high power SRAM used in cache memory
9. Data integrity and data consistency is always a problem
10. On chip cache memory is necessary for better performance, adding to cost, area and power usage
11. Difficult to predict performance bottlenecks and power usage due to probabilistic nature of the cache memory

12. Pipeline restart and branch prediction is not deterministic which causes execution delays and extra power usage

13. FPU pipeline restart is probabilistic which uses more power

14. Power usage a major problems for future generations of processors and face many problems due to excessive heat dissipation

15. Look-ahead logic and branch prediction becomes very complex

16. Real-time DSP systems for signal processing not possible with cache based systems

17. Software compatibility with previous generations not always possible

18. Server architectures have limited performance due to limitations of cache memory and problems related to inter-processor communications

NEW AND UNEXPECTED RESULTS PRODUCED BY THE NEW CACHELESS COMPUTER SYSTEM

1. Deterministic systems are always superior to probabilistic systems because they are more predictable and manageable and are easy to design

2. CPU can execute programs at CPU speed without interruption since there is no need for cache memory to supply required instruction flow and data.

3. Eliminates the need for on chip cache memory and related control and management logic needed to manage on chip cache memory, making dramatic reduction in the total wafer size and heat dissipation

4. Eliminates the need for on chip cache memory making dramatic improvements in the total power usage, heat dissipation and resulting thermal management

5. Total die size for the processor is greatly reduced due to improvements in processor pipelining schemes and control logic

6. The speed and performance is not related to size and speed of the cache memory.

7. Real time operation is possible for all the systems including the DSP

8. Real time "on the fly" program transfer to the memory and instant execution of the program due to the availability of details of program execution sequence in advance

9. Data integrity and consistency problems are eliminated because there is no need to keep two copies of data in cache memory as well as main semiconductor memory, creating conflicts between the stale data and recently updated data

10. Server architectures with multiple processors have improved performance due to elimination of problems related to inter-processor communication. In addition, problems relating to data integrity and consistency are also eliminated

11. Greatly improves fault tolerance and reliability for single processor and multiprocessor systems

12. The size of the programs that can be executed without interruption and conflicts in data integrity is equal to the size of the entire main semiconductor memory and not just the size of the cache memory

13. Pipeline restart and branch prediction is deterministic which avoids execution delays and reduces heat dissipation and power usage

14. FPU pipeline restart is deterministic which avoids execution delays and reduces heat dissipation and power usage

15. FPU pipeline depth can be optimized for each FPU operation in advance due to deterministic information available, which avoids execution delays and reduces heat dissipation and power usage

16. The power management is brought in to the deterministic domain. This means the power usage and thermal behavior can be predicted at the compile time.

17. Power is further reduced since more program execution occurs from low power DRAM than high power transition buffer

18. The power usage and thermal behavior envelope can be predicted at the compile time and queries pertaining to that behavior can be made at the compile time and necessary adjustments can be made in advance

19. The CPU speed can be reduced at the exact point in the program execution due to prior information obtained at the compile time to improve power usage and thermal management

20. Existing software can be used without modification with greater performance capabilities than obtained previously

21. Performance bottlenecks can be predicted at the compile time and necessary adjustments can be made in advance, thereby avoiding timing problems and system crashes

Prior to discussing the claims the applicant would like to discuss and analyze the references cited by the examiner and identify the areas in which the present invention has advantages and produce valuable new, improved and unexpected results.

The Mekhiel patent teaches a method of reducing memory latency in a computer system. The patent however, still uses a cache memory based probabilistic system. Lines 1-4 in abstract states, a buffer stores recently used addresses and associated data. The basis of operation of this architecture is the statistical probability that the data CPU is searching is found in collection of recently used addresses and data stored on the collection of recently used data in the buffer. If the CPU is not successful in finding this data in the buffer a cache miss occurs. This will require and the cache will be refilled with required data. Page 9 line 62 to page 10, line 17, describes the architecture and need for a buffer as well as a cache memory for this architecture. Page 10 lines 25-46 describes the mechanism of "miss" that can occur and also how a hit can occur. Detailed flow charts describe in greater detail the operation and function of this architecture and also describe how a miss on a cache memory is handled. Page 8 describes the function and the reason the buffer based system is designed. Figures 3A and 3B shows how a buffer is needed in addition to the cache memory for this system to operate. Even with extra hardware and associated execution delays, a miss can occur. The system proposed by Mekhiel does not achieve the new and unexpected results achieved by the present invention.

The limitations and disadvantages of the cache memory based probabilistic system and the solution to this problem offered by the new cacheless architecture is the subject of the present invention. There is no solution offered in the Mekhiel patent to eliminate cache misses.

The second major aspect of the present invention not offered in Mekhiel patent is the deterministic power management.

What has been suggested in Mekhiel patent is a method to improve the performance of the cache memory. The cache memory still is open to cache miss and will be needed to refill the cache if the cache miss occurs. The data integrity and consistency problems are the same as encountered in the cache based systems.

The Goodnow et al. patent teaches a method of reducing memory latency in a computer system. The patent however, still uses a cache memory based probabilistic system. There is no solution offered in the patent to eliminate cache misses.

In the abstract on line 1 it teaches a method of preloading a cache memory based on the information contained in a computer generated map created at the compile time.

Page 4 line 43 – 55 describes the purpose of the data processing architecture. Here it is indicated the need to place data and instruction in a cache by "locality of references" for CPU to execute. This process mandates a requirement that a data or instruction being searched is to be found in a particular area of the memory.

On page 5 lines 7-11 the patent refers to "hit" ratios and "miss" mechanism. This is the very nature of probabilistic systems the present invention is trying to eliminate.

Goodnow has suggested a method to improve the performance of the system by a small cache memory called program map RAM. The cache memory will be loaded with the best estimation of expected branches and data locations obtained during the compile process. This map will then be used to preload cache memory. The cache however, is still open to cache misses and the CPU will not be able to sustain continuous operation without access to main memory. This will add execution delays. It will also be necessary to refill the cache with new data if the map memory does not provide the required data.

The data integrity and consistency problems are the same as encountered in the cache based systems.

Page 8 line 52 it refers to a preloading a cache based system with a best possible estimation of future branches. Here again is a probabilistic element of chance, which the present invention is trying to eliminate.

The Young et al. paper describes the technique of reordering basic blocks of program called branch alignment. The purpose of this algorithm is to reduce misfetch penalties, a method of addressing the problems of cache memory misses, control penalties and instruction misfetch encountered in microprocessor operation. A unique case of code placement to reduce pipeline penalties due to control transfer instructions has been defined as branch alignment in this paper. The object of the paper is to investigate cache memory misses and control penalties encountered by using well-known DTST analysis tool. The assumption is made here is that there is a better method of reordering and preloading cache memory to increase hit ratio on cache memories and reduce the miss thereby prevent problems relating to cache memory misses, which is the subject matter of the present invention.
Page 184, left column line 8 it refers to misfetch penalties and problems relating to preloading a cache based system with a best possible estimation of future branches

Page 185 lines 3-6 on right column it refers to achieving a permutation with lower overall cost.

In final conclusion on page 191 the authors are intrigued by some of he results of their experimentation. They found that the results are not what they had expected. This strongly militates in favor of present invention because this represents the investigation in the understanding of probabilistic nature of cache memory based systems.

The effort here is reducing misfetch and penalties associated with cache based systems. The patent however, still uses a cache memory based probabilistic system. There is no solution offered in the patent to eliminate cache misses.

The Sen et al. paper teaches a method of addressing the problems of high latency and loss rates encountered in Internet environments. The paper is intended to offer suggestions to accomplish reduction of delay and losses and overcome throughput problems encountered in Internet transmissions. This is achieved by means of prefix caching. The important goal here is to reduce traffic between the server and the proxy

Page 1310 line 12-20 on right hand column, the paper refers to this scheme as traditional cache memory for storing text and data to be used on Internet environment. The primary goal here is to reduce network delay and losses encountered in Internet transmissions.

On page 1311 lines 12-14 on left column the paper suggest transmitting larger frames in advance of each burst. This is not possible in real time since it is necessary to determine the size and nature of data streams in real-time to accomplish the real time transmission. However, this suggested process of transmitting large frames consumes finite time, which introduces delays, thus the process cannot be accomplished in real time. In fact, if a large number of frames containing large frame content in each frame are encountered in series, the proposed system will not work altogether due to the limited bandwidth of the Internet transmission networks and limited size of the proxy buffer. This is similar to prior art cache memory based systems where a statistical probability of hit is assessed and probabilistic estimation of success is determined.

Workhead smoothing is discussed on page 1313 in detail. The function of work head smoothing is to reduce the variability of network resources requirements by transmitting large frames in advance of each burst. Figure 2 shows the smoothing model where the delay from the arrival to playout depends on many statistically probabilistic events. And

as indicated on page1313 right column, lines 5-8, this scheme offers more opportunities for statistical multiplexing gain.
While the objective of the present invention is to avoid the dependence on probabilistic events to achieve deterministic operation for real-time computing systems.

There is no concept of CPU cycle time to be matched with instruction execution time in Sen paper. The server CPU has no direct relation to the data being transmitted. The data transmission can be synchronous or asynchronous and can be handled by peripherals not dependent on CPU cycle time.

The streaming video will work best for pre-recorded video images or images with finite delay introduced in the transmission. There is no need stressed here for any concept of real time computing needed to achieve this video transfer nor there is any suggestion in this paper regarding real-time computing without cache misses or data integrity and data consistency problems. Under abstract on lines 17-19 the paper describes the objective of the proxy prefix caching, which is to serve multiple clients with multiple streams. In this case the transmission can be any video clip, that may not even be related to or part of the streaming video that might follow. The popular clips are intended to keep the viewer interested in watching the video images till the next images arrive. The streaming video transfer is only in one direction. There is no concept of two-way communication. In fact it is not possible to achieve two-way communication in the proposed architecture.

The present invention has an architecture that will prevent cache misses and problems such as system crashes and unacceptable execution delays for many applications. The Sen paper has no concept of power management or need thereof.
The present invention operates with real time communication in both the direction. The inputs to the computing environment is as important as the outputs of the system at any given time.

Also in the prefix caching there is no need and possibility of predicting performance bottlenecks at compile time. In fact there is no codes to compile or predictions to be made in pre-recorded video images.

The new and unexpected results produced by the present invention are listed above under

NEW AND UNEXPECTED RESULTS PRODUCED BY THE NEW CACHELESS COMPUTER SYSTEM

Important note: The office action has referred to claims with the old claim numbers 1-20. The new claim numbers are 21-40. To keep the consistency with the office action, the applicant has used the old claim numbers. Where applicable and necessary please substitute the new numbers with the old numbers.

CLAIM OBJECTIONS

The new claims 21-40 will overcome the objections raised in the office action. The mistakes have been corrected.

CLAIM REJECTION – 35 USC 112

Claims 4-8 and 11-12 are rejected under 35 USC 112, first paragraph, as failing to comply with the enablement requirement. The claims contains subject matter which was not described in the specification in such a way as to enable one skilled in art to which it pertains, or which it is most nearly connected, to make and/or use the invention.

Claims 1-20 are rejected under 35 USC 112, second paragraph, as being indefinite for failing to particularly point out distinctly claim the subject matter which applicant regards as the invention.
The terms low-cost, low power, and slower access time in claims 1-20 are relative terms, which render the claims indefinite.

On page 4 and page 40 of the specification the relative comparison between the access time and speed of the DRAM and SRAM memories have been presented.

The applicant would like to respectfully point out that one skilled in art would readily recognize the difference between DRAM used in main memory and SRAM used in the transition buffer. The cost comparison and the relative power usage are well known facts to be found in the available literature in the art. The access time of DRAM is greater than that of SRAM. This fact is well known in the prior art. Thus the meaning as to the slow or high-speed capabilities of memories is well known in prior art.

To comply with the requirements of 35 USC 112, the applicant has amended the specification and additional information given in the "conclusion and summary of advantages" added on page 46 of the specification will give better understanding of and meaning to these terms.

The specification is amended to include description of logic in CPU that allows the storing of recently used data in the transition buffer. Description of logic in CPU that implements pipelined storage is also included in the amendment. The amended specification also includes description relating to storage area in CPU for storing data that available to CPU and is related to main execution memory. Three descriptions are added on page 25 of the specifications.

The new claims 21-40 will correct the deficiency regarding "means" clause.

Claim 8 is incomprehensible. Accordingly prior art is not applied to this claim.

Claim 8 claims a pipelined storage that is implemented in the transition buffer similar to the one implemented in the main execution memory by CPU using a special logic to implement multiway branches without execution delays. This logic gives the CPU access to the instructions the CPU needs for implementing special branch instructions to execute the branch instructions without any execution delays.

Claim 13 is incomprehensible. Accordingly prior art is not applied to this claim.

Claim 13 refers to a pipelined storage that is implemented in the main execution memory by CPU using a special logic to implement multiway branches without execution delays. A special logic is also designed for the CPU to implement the pipelined storage in the transition buffer when needed. This logic gives the CPU access to the instructions the CPU needs for implementing special branch instructions to execute the branch instructions without any execution delays. Claim 13 claims logic in CPU that decodes the instructions located in pipeline storage in main execution memory or transition buffer in advance. These instructions are obtained from the pipelines implemented in main execution memory or the transition buffer

CLAIM REJECTION – 35 USC 103

Claims 1-6, 9-20 are rejected under 35 USC 103(a) as being unpatentable over Mekhiel (US6, 587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999).

As in claim 1,
Where it is readily apparent in the above steps that the data from both the buffer and main memory comprises.. new fetch is required. Further regarding claim 1… as taught by Sen.

With regard to the proposed combination of Mekhiel, Goodnow and Sen, it is well known that in order for any prior art references themselves to be validly combined for use in a prior-art USC 35, 103 rejection, the references themselves or some other prior art must suggest they be combined as was stated in **In re Serneker**, 217 USPQ 1.6 (CAFC 1983): prior art references in combination do not make an invention obvious unless something in the prior art references would suggest the advantages to be derived from combining their teachings.

The cache memory based computer system is a probabilistic entity. All probabilistic systems eventually fail in performing its intended task, which is required of the probabilistic systems. The present invention transforms the prior art cache memory based computer system from the probabilistic system to deterministic system.

Cache memory based systems operating as probabilistic system has finite success ratio. The cache will fail in obtaining required data or instruction during certain operating conditions. Even with a very high hit ratio of 99%, failure can lead to major operational and data integrity problems.

This failure will lead to refill of cache memory that requires a long time interval to achieve the refill. During this time period, serious problems can occur such as execution delays, loss of input values and a system crash can occur.
The difference between the present invention and the prior art computer systems are in every area of design operation and performance.

The applicant would like to submit that the steps described by Mekhiel are very much in line with prior art teachings. The effort made here is to improve the performance of the cache memory by introducing a buffer. As described earlier in the analysis of Mekhiel patent, the purpose here is to improve the function of cache memory and not to eliminate the cache memory.

The basis of operation of this architecture is the statistical probability that the data CPU is searching is found in collection of recently used addresses and data stored in the buffer. The applicant would like to point out that this is not the 100 percent of the information that is needed for CPU to operate without a cache memory "miss". If the requested data is found on the buffer in the collection of recently used data in the buffer, the buffer fills the cache memory in burst mode. If the CPU is not successful in finding this data in the buffer, a cache "miss" occurs. This will require and the cache will be refilled with

required data. Page 9 line 62 to page 10, line 17, describes the architecture and need for a buffer as well as a cache memory for this architecture.

The present invention is about eliminating use, function and purpose of the cache memory in a computer system.

Another important goal of the present invention is to introduce better power management in deterministic environment. The applicant would like to emphasize the 18 limitations of prior art cache memory based system described and stated earlier in this office action. The teachings of Mekhiel do not eliminate these limitations, in fact this indicates that the problems of cache memory miss and associated limitations are considered unsolvable. The architecture proposed in the present invention produce valuable new, improved and unexpected results.

Goodnow teaches a computer system where best estimation of expected instructions and data to be executed are determined at stored in a small cache memory called map RAM. The applicant would like to point out that this is not 100 percent of the information that is needed for CPU to operate without a cache memory "miss". This data is then used to preload cache memory.

The cache however, is still open to cache misses and the CPU will not be able to sustain continuous operation without access to main memory. This will add to execution delays. It will be necessary to refill the cache with new data if the map memory does not provide the required data. The data integrity and consistency problems are the same as encountered in the cache based system.

Sen teaches a system of delivering large multimedia objects with reduced network delay and losses.
Page 1310 line 12-20 on right hand column, the paper refers to this scheme as traditional cache memory for storing text and data to be used on Internet environment. The primary

goal here is to reduce network delay and losses encountered in Internet transmissions. There is no concept of real-time computing suggested here.

Page 1311 lines 12-14 on left column the paper suggest transmitting larger frames in advance of each burst. This is not possible in real time since it is necessary to determine the size and nature of data streams in real-time to accomplish real-time transmission. But this suggested process of transmitting large frames consumes finite time, which introduces delays, thus the process cannot be accomplished in real-time. In fact, if a large number of frames containing large frame content in each frame are encountered in series, the proposed system will not work due to the limited bandwidth of the Internet transmission networks and limited size of the proxy buffer. This is similar to prior art cache memory based systems where a statistical probability of hit is assessed and probabilistic estimation of success is determined.

And, as indicated on page1313 right column, lines 5-8, this scheme offers more opportunities for statistical multiplexing gain. This is again a probabilistic estimation.

However, The objective of the present invention is to avoid the dependence on probabilistic events to achieve deterministic operation for real-time computing systems.

There is no concept of CPU cycle time to match instruction execution time in Sen paper. The server CPU has no direct relation to the data being transmitted. The data transmission can be synchronous or asynchronous and can be handled by peripherals not dependent on CPU cycle time.

The primary objective of the present invention is to bridge the gap between the CPU instruction execution time and memory access time thus eliminating the need for cache memory. The architecture proposed in the present invention produce valuable new, improved and unexpected results.

The examiner writes on page 9 lines 4-10 of this office action " Regarding claim 1, it would have been obvious to one of ordinary skill in art at the time of invention by applicant to store starting locations of program branch instructions .....as taught by Goodnow".

The applicant would like to respectfully point out that both Mekhiel and Goodnow teaches a system that uses cache memory based probabilistic system. These systems are subject to cache memory "miss" similar to prior art systems. The effort is directed toward improving the performance of the cache, not eliminating the disadvantages of cache. This militates in favor of the preset invention because the problems related to cache memory are considered unsolvable. Two probabilistic systems subject to miss problems cannot possibly be used to conceive a cacheless deterministic system.

The examiner writes on page 9 lines 11-18 of this office action "Further regarding claim 1, the teachings of Sen would suggest one of ordinary skill in the art that a stream of program data retrieved form main memory could be .....in order to conserve space in the cache as taught by Sen "

The applicant would like to respectfully point out that the purpose of the Sen paper is to teach method of reducing losses and reduce latency in transmission. A digital computer cannot accept any loss in data. In digital computer, the data integrity problems are caused by delay in updating data and not loss of any data. The latency involved in operation of computer is many orders different than the latency between an Internet server and proxy or client.

Two probabilistic cache memory based systems taught by Mekhiel and Goodnow combined with a system taught by Sen to transmit data that may not be transmitted in real time cannot possibly be used to conceive a cacheless deterministic system that can perform real-time operations. There is an unobvious difference between three probabilistic cache based systems combined to produce a system that is not operative and a system proposed by the applicant which is deterministic system that solves many

problems of prior art cache based systems that were considered unsolvable and produces new and unexpected results.

The applicant would like to refer to the important disadvantages of prior art computer system listed earlier in this office action under:

LIMITATIONS OF CACHE MEMORY BASED SYSTEMS

These disadvantages are still very much part of the prior art systems. In absence of any superior systems, the prior art computers still operate with the limitations listed above. This militates in favor of the preset invention because the problems related to cache memory are considered unsolvable.

The present invention solves the problems of prior art computer systems with novel concept of deterministic and real-time computing. The important features that make the present invention unobvious and hence patentable are listed earlier in this office action under:

NEW AND UNEXPECTED RESULTS PRODUCED BY THE NEW CACHELESS COMPUTER SYSTEM

The applicant would like to cite two important decisions by CAFC.

1. Orthopedic equipment co. v. United States, 217 USPQ 193. 1999 (CAFC 1983).
2. Uniroral Inc.v. Rudkin-Wiley corp., 5 USPQ 2d 1434 (CAFC 1988).

That the suggestion to combine the references should not come from applicant was forcefully stated in Orthopedic equipment co. v. United States, 217 USPQ 193. 1999 (CAFC 1983):

It is wrong to use the patent in suit [here patent application] as guide through the maze of prior art references, combining the right reference in the right way to achieve the result of the claims in suit [here claims pending]. Monday morning quarterbacking is quite improper when resolving the question of in a court of law [here the PTO].

As was further stated in here prior-art references require selective combination by the to render obvious a subsequent invention, there must be some reason for the combination

other than the hindsight gleaned from the invention itself. .... something in the prior art must suggest the desirability and thus the obviousness of making the combination.

This was further affirmed by the Board in Ex parte Levengood. 28 USPQ. 2d 1300 (PTO BA&I. 1983).

In order to establish a pima facie evidence of obviousness, it is necessary for the examiner to present evidence, preferably in the form of some teaching, suggestion, incentive or inference in the applied prior art, or in the form of generally available knowledge that one having ordinary skill in the prior art would have been led to combine the relevant teachings of the applied reference in the proposed manner to arrive at the claimed invention. That which is within the capabilities of one skilled in the art is not synonymous with obviousness. ....that one can reconstruct and/or explain the theoretical mechanism of an invention by means of logic and sound scientific reasoning does not afford the basis for obviousness conclusion unless that logic and reasoning also supplies sufficient impetus to have led one of ordinary skill in the art to combine the teachings of the references to make the claimed invention. ... our reviewing courts have often advised the Patent and Trademarks office that it can satisfy the burden of establishing a prima facie case of obviousness only by showing some objective teaching in either in the prior art, or knowledge generally available to one of having ordinary skill in the art, that would lead that individual to combine the relevant teachings of the reference. Accordingly, an examiner cannot establish obviousness by locating references, which describe various aspects of a patent applicant's invention without also providing evidence of the motivating force, which would impel one skilled in the art to do what the patent applicant has done.

In the present case, there is no reason given in the office action to support the proposed combination, other than the statement "to store starting memory location .....as taught by Goodnow in the system of Mekhiel where the buffer in Goodnow corresponds to buffer in the system of Mekhiel ....as taught by Goodnow". However, the fact that

Goodnow teaches a buffer to store memory locations is not sufficient to gratuitously and selectively substitute parts of one reference for a part of another reference in order to meet applicant's novel claimed combination.

Also in the present case, there is no reason given in the office action to support the combination of Sen, Goodnow and Mekhiel, other than the statement " the teaching of Sen would suggest to one of ordinary skill in art that a stream of program data .....while conserving buffer space. Therefore it would be obvious to one of ordinary skill in art at the time of invention by the applicant to fetch .....as suggested by the teachings of Sen, in the system made obvious by the combination of Mekhiel and Goodnow in order to conserve space in the cache taught by Sen". However, the fact that Sen teaches method of transferring data objects over internet and Mekhiel teaches a buffer to store most recently used addresses and associated data and Goodnow teaches a buffer to store memory locations is not sufficient to gratuitously and selectively substitute parts of one reference for a part of another reference in order to meet applicant's novel claimed combination.

As stated in above Levengood case,
That one can reconstruct and/or explain the theoretical mechanism of an invention by means of logic and sound scientific reasoning does not afford the basis for obviousness conclusion unless that logic and reasoning also supplies sufficient impetus to have led one of ordinary skill in the art to combine the teachings of the references to make the claimed invention. ...

MPEP 2143 states, to establish a prima facie case of obviousness there basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or combine reference teachings. Secondly, there must be reasonable expectation of success. Finally, the prior art reference or (references when combined) must teach or suggest all the claim limitations.

The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art, not in applicant's disclosure. In re Vaeck, 947 F.2d 488, 20 USPQ2d 1438(Fed. Cir. 1991).

There is no justification in Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., or any other prior art reference independent of applicants disclosure which suggests that these references be combined much less be combined in a manner proposed.

In addition, the proposed combination would not be physically possible or operative.

Even if the combinations of Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., be combined in a manner proposed, the proposed combination would not show all of the novel features of the present invention.

The new design concepts discovered by the applicant produce the new and unexpected results for the computer system; hence the concepts and resulting embodiments of the present invention are unobvious and patentable over these prior art references.

There is no suggestion in the three references, either in the form of some teaching, suggestion, incentive or inference in the applied references, or in the form of generally available knowledge that one having ordinary skill in the prior art would have been led to combine the relevant teachings of the applied reference in the proposed manner to arrive at the claimed invention and resolve the issue of unobviousness.

The applicant therefore submits that combining Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., is not legally justified and is therefore

improper. Therefore the applicant submits the rejection on these references is also improper and should be with drawn.

As in claim 2, Mekhiel discloses that the main memory is comprised of ..... Buffer using SRAM.

The applicant submits that claim2 is a dependent claim. The dependent claim 2 incorporates all the subject matter of claim 1 and add additional subject matter which makes it fortiori and independently patentable over the cited references. The claim 2 is defined as having the basic elements of claim 1 which uses the novel concept of transition buffer to achieve deterministic operation without a cache memory and attendant cache memory miss problems. The prior art teachings proposed by Mekhiel does not use deterministic operation of transition buffer to obtain low power execution of instructions and the data consistency and cache coherency function as described in claims 1 and 2.

The applicant would like to point out that the use of SRAM as a transition buffer is not same as buffer taught by Mekhiel. The transition buffer operates only during the execution of starting locations compared to buffer in Mekiel, which operates more frequently. Two important goals of the present invention are bridging the gap between CPU instruction execution time and memory access time and reduction in total power usage by the system. Because the SRAM operates only during starting locations, higher speed SRAM can be used without power penalty to obtain CPU instruction execution time equal to memory access time and at the same time keep power usage low.

In prior art systems as taught by Mekhiel, most of the instruction execution occurs from the cache memory comprising of SRAM and the main DRAM is used mainly to store the program for the cache SRAM to execute.

A novel and important feature of the present invention is that most of the execution is carried out in the DRAM. The DRAM can be put in desired parallel banks without speed penalty and obtain low power advantage not found in prior art systems.
As stated in claim 2, new and unexpected results are produced when the system according to the present invention is designed with low power DRAM in main memory and high speed SRAM with lower usage during instruction execution to reduce total power usage in the system.
There is no justification in Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., or any other prior art reference independent of applicants disclosure which suggests that these references be combined much less be combined in a manner proposed.

In addition, the proposed combination would not be physically possible or operative.

Even if the combinations of Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., be combined in a manner proposed, the proposed combination would not show all of the novel features of the present invention.

The new design concepts discovered by the applicant produce the new and unexpected results for the computer system; hence the concepts and resulting embodiments of the present invention are unobvious and patentable over these prior art references.

There is no suggestion in the three references, either in the form of some teaching, suggestion, incentive or inference in the applied references, or in the form of generally available knowledge that one having ordinary skill in the prior art would have been led to combine the relevant teachings of the applied reference in the proposed manner to arrive at the claimed invention and resolve the issue of unobviousness.

The applicant therefore submits that combining Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., is not legally justified and is therefore improper. Therefore the applicant submits the rejection on these references is also improper and should be with drawn.

Regarding claims 4-6, relying on the rationale for the rejection of claim1, ...with buffer (Figure 3, elements labeled "L1 CACHE" and "L2 CACHE").

The applicant would like to respectfully request that the rejection of claims 4-6 should be withdrawn on the same basis as requested for claim 1.

The applicant submits that claims 4-6 are dependent claims. The dependent claims 4-6 incorporates all the subject matter of claim 1 and add additional subject matter which makes it fortiori and independently patentable over the cited references. The claims 4-6 is defined as having the basic elements of claim 1 which uses the novel concept of transition buffer to achieve deterministic operation without a cache memory and attendant cache memory miss problems. The prior art teachings proposed by Mekhiel and Goodnow does not use deterministic operation of transition buffer to obtain the data consistency and cache coherency function and use of recently used data as described in claims 1 and claims 4-6.

The applicant would like to respectfully point out that the transition buffer is not used to store only recently used data. The object of the transition buffer is to store starting locations of all the instructions and incoming data or outgoing data for the entire program.

Primary objective of this invention is to eliminate the function, need, use and effect of cache memory used in the prior art computer systems.

The present invention will function without the recently used data while the Mekhiel and Goodnow systems will not operate without the recently used data stored in the buffer. The function and use of buffer in the present invention and the Mekhiel and Goodnow system are very different.
The recently used data in the present invention is employed to reduce power usage by minimizing access to both the main memory and the transition buffer for data used more frequently.

As stated in claims 4-6, new and unexpected results are produced when the system according to the present invention is designed with goal of reducing total power usage by storing recently used data in the transition buffer.

There is no justification in Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., or any other prior art reference independent of applicants disclosure which suggests that these references be combined much less be combined in a manner proposed.

In addition, the proposed combination would not be physically possible or operative.

Even if the combinations of Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., be combined in a manner proposed, the proposed combination would not show all of the novel features of the present invention.

The new design concepts discovered by the applicant produce the new and unexpected results for the computer system; hence the concepts and resulting embodiments of the present invention are unobvious and patentable over these prior art references.

There is no suggestion in the three references, either in the form of some teaching, suggestion, incentive or inference in the applied references, or in the form of generally

available knowledge that one having ordinary skill in the prior art would have been led to combine the relevant teachings of the applied reference in the proposed manner to arrive at the claimed invention and resolve the issue of unobviousness.

The applicant therefore submits that combining Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., is not legally justified and is therefore improper. Therefore the applicant submits the rejection on these references is also improper and should be with drawn.

As in claims 11-12, relying on the rationale for the rejection of claim1, ...with buffer of Mekhiel by definition is a storage area that stores data available to CPU ....determined at compile time as required by claim 12.
The applicant submits that claims 11-12 are dependent claims. The dependent claims 11-12 incorporates all the subject matter of claim 1 and add additional subject matter which makes it fortiori and independently patentable over the cited references. The claims 11-12 are defined as having the basic elements of claim 1 which uses the novel concept of transition buffer to achieve deterministic operation without a cache memory and attendant cache memory miss problems. The prior art teachings proposed by Mekhiel and Goodnow does not use deterministic operation of transition buffer to obtain the data consistency and cache coherency function and use of CPU to store data as described in claims 1 and 11-12.

The applicant would like to respectfully point out that the CPU is not used to store data for execution of all the instructions. the transition buffer is used to store starting locations for CPU to operate. The object of the transition buffer is to store starting locations of all the instructions and incoming data or outgoing data for the entire program.

Primary objective of this invention is to eliminate the function, need, use and effect of cache memory used in the prior art computer systems.

The present invention will function without the data stored in the CPU, determined at the compile time. While the Mekhiel system will not operate without the data stored in the buffer. And the Goodnow system will not operate without the data stored in the MAP storage area as described. The goal of 100 percent hit rate in Goodnow system is never achieved because it is a probabilistic system. The function and use of information stored in CPU in the present invention and the Mekhiel and Goodnow system are very different.

The data stored in CPU in the present invention is employed to reduce power usage and further improve performance by minimizing access to both the main memory and the transition buffer for data used more frequently.

As stated in claims 11-12, new and unexpected results are produced when the system according to the present invention is designed with goal of reducing total power usage by storing data in the CPU.

There is no justification in Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., or any other prior art reference independent of applicants disclosure which suggests that these references be combined much less be combined in a manner proposed.

In addition, the proposed combination would not be physically possible or operative.

Even if the combinations of Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., be combined in a manner proposed, the proposed combination would not show all of the novel features of the present invention.

The new design concepts discovered by the applicant produce the new and unexpected results for the computer system; hence the concepts and resulting embodiments of the present invention are unobvious and patentable over these prior art references.

There is no suggestion in the three references, either in the form of some teaching, suggestion, incentive or inference in the applied references, or in the form of generally available knowledge that one having ordinary skill in the prior art would have been led to combine the relevant teachings of the applied reference in the proposed manner to arrive at the claimed invention and resolve the issue of unobviousness.

The applicant therefore submits that combining Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., is not legally justified and is therefore improper. Therefore the applicant submits the rejection on these references is also improper and should be with drawn.

Claim 15 is rejected using the same rationale as for the rejection of claim1, ...where it is noted that Goodnow clearly contemplates an interrupt ....in a program execution flow (column 2, line 44 to column 3, line 18).

The applicant submits that claim 15 is a dependent claim. The dependent claim 15 incorporates all the subject matter of claim 1 and add additional subject matter which makes it fortiori and independently patentable over the cited references. The claim 15 is defined as having the basic elements of claim 1 which uses the novel concept of transition buffer to achieve deterministic operation without a cache memory and attendant cache memory miss problems. The prior art teachings proposed by Goodnow does not use deterministic operation of transition buffer to obtain zero latency interrupt response for all the interrupt conditions and the data consistency and cache coherency function as described in claims 1 and 15.

The applicant would like to respectfully point out that the transition buffer is used to store the starting locations of all the interrupt service routines. The objective of the transition buffer is to store starting locations of all the interrupt service routines, instructions and incoming data or outgoing data for the entire program.

Primary objective of this invention is to eliminate the function, need, use and effect of cache memory used in the prior art computer systems.

The CPU logic to service and activate zero latency interrupt response system that activates interrupt service routine without any delay involved in accessing the main execution memory by using transition buffer to start the interrupt service routine provides deterministic response to all the interrupt service requests without causing CPU stalls.

Goodnow in column 3, line 44 to column 3, line 45 describes the probabilistic nature of interrupt service concept and possible logic to be used to fill the cache memory with appropriate data to reduce interrupt service failures. There is no solution offered to provide deterministic 100 percent responses to all the interrupt requests. What is suggested is mere improvement over prior art probabilistic system and the result is still another probabilistic system to address interrupt service problem.

The problem described by the Goodnow describes how important the present invention is. The present invention produces new and unexpected results not achieved by the prior art systems.

There is no justification in Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE,

March 1999)., or any other prior art reference independent of applicants disclosure which suggests that these references be combined much less be combined in a manner proposed.

In addition, the proposed combination would not be physically possible or operative.

Even if the combinations of Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., be combined in a manner proposed, the proposed combination would not show all of the novel features of the present invention.

The new design concepts discovered by the applicant produce the new and unexpected results for the computer system; hence the concepts and resulting embodiments of the present invention are unobvious and patentable over these prior art references.

There is no suggestion in the three references, either in the form of some teaching, suggestion, incentive or inference in the applied references, or in the form of generally available knowledge that one having ordinary skill in the prior art would have been led to combine the relevant teachings of the applied reference in the proposed manner to arrive at the claimed invention and resolve the issue of unobviousness.

The applicant therefore submits that combining Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., is not legally justified and is therefore improper. Therefore the applicant submits the rejection on these references is also improper and should be with drawn.

Regarding claim 17, it is noted that claim 17 differs from claim 1 in two aspects. First, claim 17 includes limitations directed towards writing data that were not present in claim 1. Secondly, the combined instruction and data paths....to the Harvard architecture.

As to the first aspect of the claim 17, Mekhiel teaches that the CPU .....remaining data to the main memory(figure 6, steps300, 390, 460, 480 and 490; column 12, lines 36-59).

The applicant would like to respectfully submit that there is a major difference between the Harvard architecture of prior art systems and the architecture presented in the present invention.

The applicant would like to submit that the steps described by Mekhiel are very much in line with prior art teachings. The effort made here is to improve the performance of the cache memory by introducing a buffer. As described earlier in the analysis of Mekhiel patent, the purpose here is to improve the function of cache memory and not to eliminate the cache memory.

The basis of operation of this architecture is the statistical probability that the data CPU is searching is found in collection of recently used addresses and data stored in the buffer. The applicant would like to point out that this is not 100 percent of the information that is needed for CPU to operate without a cache memory "miss". If the requested data is found on the buffer in the collection of recently used data in the buffer, the buffer fills the cache memory in burst mode. If the CPU is not successful in finding this data in the buffer, a cache "miss" occurs. This will require and the cache will be refilled with required data. Page 9 line 62 to page 10, line 17, describes the architecture and need for a buffer as well as a cache memory for this architecture.

Figure 6, step 300 requires that the needed memory address be available for instruction to execute. The buffer only includes recently used address and data. If the required memory address in not available in the buffer and write access is needed than according to figure 6, steps 310 and 330, instruction execution stops and main memory access is required which requires a long access times. This is not acceptable under requirements of many computing systems and definitely not acceptable for real-time computing. In fact even if the required address is found in the buffer, according to figure 6, steps 390 and 430, the instruction execution can stop. This can occur because if the required address of data

used in the execution of following instructions is not found in the buffer, the instruction execution stops and main memory access is required which requires a long access times. This is due to the fact that the buffer contains only recently used data with limited buffer space.

The Harvard architecture can operate with or without cache memory or buffer. Many applications of Harvard architecture involve real-time operations. The prior art cache memory based Harvard architectures have not been used with real time applications because the prior art logic is based on probabilistic assumption that available instruction and data present in the cache memory is the real time information needed to perform the necessary execution of instructions. This is not possible and hence delay and error in execution occurs.

A reference is made in the background section of the specification on the problems faced by the real-time systems using DSP. An article in "IEEE Spectrum" June 2001 pages 62 – 68, describes the problem in more detail. Different DSP designs using Harvard architecture have been in use for a long time. However most systems are used for real-time computing environments. Probabilistic cache memory based systems cannot be used because it cannot operate in real-time environments. The problem of using large memory systems using low power and without cache memory was considered unsolvable. The present invention offers solution to the problems considered unsolvable by the prior art systems.

In real-time computing environments there is no provision provided for time needed for cache to refill if a cache miss occurs. If time is lost in refilling the cache memory in the event of a miss, real-time data and instructions are lost. In addition, real time systems cannot operate with data integrity problems introduced by the cache coherency and consistency problems. The data updating has to occur in real-time before new real-time data arrives

Because of this, the storage size of the high-speed data and program memories operating at real-time speeds are limited due to power usage, heat dissipation and cost constraint.

Therefore, the capabilities of programs on Harvard architectures have been limited due to limitations of memories.

The Harvard architecture of the present invention uses the novel concept of transition buffer to achieve deterministic operation without a cache memory and attendant cache memory miss problems. The capacity and speed of the data and program memory can also be increased to desired sizes and values. Again, this can be achieved with low power consumption and at a lower cost.

The deterministic execution of branch and jump instructions is important for real time execution of programs. This is done without execution delays or real-time data error in the present invention.

However, in prior art systems according to Mekhiel (US6,587,920), Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999), there is no deterministic conclusion that branch and jump instructions can be implemented in real-time without data integrity problems due to cache coherency or consistency.

The new and unexpected result produced here is that a real-time computing logic and device is prevented from spending time on an operation that will not succeed. This will avoid undesirable execution delay, which further adds to the higher performance of the architecture.

The new design concepts discovered by the applicant produce the new and unexpected results for the Harvard architecture, hence the concepts and resulting embodiments of the present invention are unobvious and patentable over these prior art references.

As to the second aspect of claim 17, see Examiner's official notice below.

Examiner takes official notice of the following well-known teachings in the art.

Further regarding claim 17, the Harvard architecture is a well known architecture in the art by which the instruction and data paths are separated to include separate memories for instructions and data.... therefore it would have been obvious to one ..... made obvious by the combination of Mekhiel (US6,587,920), Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999), in order to access instructions and data simultaneously.

The purpose and differences between the Harvard architecture of prior art systems and the architecture presented in the present invention have described in detail in the explanation given earlier on the remarks for claim 17.

Emphasizing the novel features of the deterministic Harvard architecture, new and unexpected results are produced. The Harvard architecture of the present invention uses the novel concept of transition buffer to achieve deterministic operation without a cache memory and attendant cache memory miss problems. The capacity and speed of the program and data memory can also be increased to desired sizes and values. Again, this can be achieved with low power consumption and at a lower cost.

The deterministic execution of branch and jump instructions is important for real time execution of programs. This is done without execution delays or real-time data error in the present invention.

The architecture of the present invention goes beyond separate data and instruction paths and memories. The use of transition buffer along with deterministic information available at compile time allows real-time operation with large program and data memories. This feature is not found in prior art systems. Also achieved is power usage advantage not available in prior art Harvard architecture based systems. These features of the new architecture will open the DSP systems to new applications not available before.

There is no justification in Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., or any other prior art reference independent of applicants disclosure which suggests that these references be combined much less be combined in a manner proposed.

In addition, the proposed combination would not be physically possible or operative.

Even if the combinations of Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., be combined in a manner proposed, the proposed combination would not show all of the novel features of the present invention.

The new design concepts discovered by the applicant produce the new and unexpected results for the computer system; hence the concepts and resulting embodiments of the present invention are unobvious and patentable over these prior art references.

There is no suggestion in the three references, either in the form of some teaching, suggestion, incentive or inference in the applied references, or in the form of generally available knowledge that one having ordinary skill in the prior art would have been led to combine the relevant teachings of the applied reference in the proposed manner to arrive at the claimed invention and resolve the issue of unobviousness.

The applicant therefore submits that combining Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., is not legally justified and is therefore improper. Therefore the applicant submits the rejection on these references is also improper and should be with drawn.

Examiner takes official notice of the following well-known teachings in the art: Regarding claim 3, interleaving access among parallel groups of memory cell.... System made obvious by Mekhiel, Goodnow and Sen.

The applicant submits that claim 3 is a dependent claim. The dependent claim 3 incorporates all the subject matter of claim 1 and add additional subject matter which

makes it fortiori and independently patentable over the cited references. The claim 3 is defined as having the basic elements of claim 1 which uses the novel concept of transition buffer to achieve deterministic operation without a cache memory and attendant cache memory miss problems. The prior art teachings does not use of interleaved memory to obtain the function as described in claims 1 and 3.

Interleaved memory has been used in prior art to hide latency in memory. The function of interleaved memory has been to provide access and transfer to only consecutive locations of memory, generally in a burst mode.
Interleaved memory cannot execute a jump or branch instruction without a delay in execution. If the CPU requires an access to a non-sequential location on an interleaved memory to implement instruction such as a branch or jump, an execution delay equal to access time of the next group of memory locations will be introduced. This will make the interleaved memory as taught in prior art and as described in the official notice incapable of handling branch instructions. The implementation of deterministic multiway branch instructions without execution delay are one of the strengths of the present invention. The prior art architecture of the interleaved memory to hide latency will not be able to implement the multiway branch without cache memory miss if the required location is not found on the cache memory. If the interleaved memory can allow CPU to execute jump and branch instructions, there will be no need for cache memory or any other architecture to fill the gap between the CPU cycle time and the memory access time as has been used in prior art.

The new architecture and design concepts discovered by the applicant produce the new and unexpected results for the computer system, hence the concepts and resulting embodiments of the present invention are unobvious and patentable over these prior art references.
There is no justification in Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., or any other prior art reference independent of applicants disclosure which suggests that these references be combined much less be combined in a manner proposed.

In addition, the proposed combination would not be physically possible or operative.

Even if the combinations of Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., be combined in a manner proposed, the proposed combination would not show all of the novel features of the present invention.

The new design concepts discovered by the applicant produce the new and unexpected results for the computer system; hence the concepts and resulting embodiments of the present invention are unobvious and patentable over these prior art references.

There is no suggestion in the three references, either in the form of some teaching, suggestion, incentive or inference in the applied references, or in the form of generally available knowledge that one having ordinary skill in the prior art would have been led to combine the relevant teachings of the applied reference in the proposed manner to arrive at the claimed invention and resolve the issue of unobviousness.

The applicant therefore submits that combining Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., is not legally justified and is therefore improper. Therefore the applicant submits the rejection on these references is also improper and should be withdrawn.

Regarding claim 5, a non-pipelined architecture is well known in the art as a low complexity design, ... in a system made obvious by the combination of Mekhiel (US6,587,920), Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999).
The applicant submits that claim 5 is a dependent claim. The dependent claim 5 incorporates all the subject matter of claim 1 and add additional subject matter which

makes it fortiori and independently patentable over the cited references. The claim5 is defined as having the basic elements of claim 1 which uses the novel concept of transition buffer to achieve deterministic operation without a cache memory and attendant cache memory miss problems. The prior art teachings does not suggest use of compile time information to predict pipeline stalls and restart requirements to obtain the function as described in claims 1 and 5.

The purpose and differences between the non-pipelined architecture of the present invention and architecture of prior art systems have been described in detail in the explanation given in the specification.

Emphasizing the novel features of the deterministic architecture of the present invention, new and unexpected results are produced. The architecture of the present invention uses the novel concept of transition buffer to achieve deterministic operation without a cache memory and attendant cache memory miss problems. In the present invention due to deterministic operation of the computer system, it possible to predict at compile time, pipeline stalls that can occur. It is also possible to predict restart events in advance. Non-pipelined architecture is very efficient in situations where the number of steps to complete instruction is small. There are many applications where simple implementation of Non-pipelined architecture can be used with advantage to achieve higher performance and low power usage. However, if there is any dependencies involved such as in complex algorithms the architecture will not produce expected results. Execution stalls can occur. The deterministic operation of the computer system can be used to predict stalls at compile time. This can allow use of simple instructions such as used in the RISC architectures. The prior art systems using non-pipelined architectures are limited by the probabilistic behavior of the data they can handle. Again, in the new architecture, compile time information can be used to achieve low power consumption at a lower cost. The deterministic execution of branch and jump instructions is important for real time execution of programs. This is done without execution delays or real-time data error in the present invention in no-pipelined operation.

The architecture of the present invention goes beyond simple implementation of non-pipelined architecture. The use of transition buffer along with deterministic information available at compile time allows real-time operation even with complex instructions. This feature is not found in prior art systems. Also achieved is power usage advantage not available in prior art systems.

There is no justification in Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., or any other prior art reference independent of applicants disclosure which suggests that these references be combined much less be combined in a manner proposed.

In addition, the proposed combination would not be physically possible or operative.

Even if the combinations of Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., be combined in a manner proposed, the proposed combination would not show all of the novel features of the present invention.

The new design concepts discovered by the applicant produce the new and unexpected results for the computer system; hence the concepts and resulting embodiments of the present invention are unobvious and patentable over these prior art references.

There is no suggestion in the three references, either in the form of some teaching, suggestion, incentive or inference in the applied references, or in the form of generally available knowledge that one having ordinary skill in the prior art would have been led to combine the relevant teachings of the applied reference in the proposed manner to arrive at the claimed invention and resolve the issue of unobviousness.

The applicant therefore submits that combining Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., is not legally justified and is therefore

improper. Therefore the applicant submits the rejection on these references is also improper and should be with drawn.

Regarding claim 6, a pipelined architecture comprising an instruction queue is well known in the art as providing higher performance than a non-pipelined architecture a low complexity design... in the system made obvious by the combination of Mekhiel (US6,587,920), Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999).

The purpose and differences between the pipelined architecture and architecture of prior art systems and the architecture presented in the present invention have described in detail in the explanation given in the specification.

Emphasizing the novel features of the deterministic architecture of the present invention, new and unexpected results are produced. The architecture of the present invention uses the novel concept of transition buffer to achieve deterministic operation without a cache memory and attendant cache memory miss problems. In the present invention due to deterministic operation of the computer system, it is possible to predict at compile time, pipeline stalls that can occur. Pipelined architecture is very efficient in situations where a large number of steps are required to complete an instruction. There are many applications where simple implementation of Non-pipelined architecture will not be successful. The pipelined architecture can be used with advantage to achieve higher performance and low power usage in complex instructions requiring multiple steps. Due to availability of required data at all the time, the prior art problems of pipeline stalls are eliminated. The problem of dependencies is handled very efficiently. There is no solution for errors related to dependencies in prior art systems. Deeper pipelines with its advantages in many areas can be difficult to restart. It is possible to fill the pipelines unnecessarily due to an incoming jump or branch instruction that will not use the pipeline. In view of high performance requirements of the present architecture over prior art, the restart problems cannot be allowed to degrade the performance. The deterministic operation of the computer system can be used to predict restart problems at compile time.

This will prevent unnecessary start of the pipelines that will not be used. It will also predict the depth of the pipeline the instructions will use. This becomes very important in pipelines for math coprocessors or FPU. Advance estimation of depth needed to execute a given instruction can improve performance and reduce power. The prior art systems using pipelined architectures are limited by the probabilistic behavior of the data, which can cause instruction stalls due to many events. Again, in the present invention this can be prevented with low power consumption and at a lower cost.

The deterministic execution of branch and jump instructions is important for real time execution of programs. This is done without execution delays or real-time data error in the present invention.

The architecture of the present invention goes beyond simple implementation of pipelined architecture. The use of transition buffer along with deterministic information available at compile time allows real-time operation even with complex instructions. This feature is not found in prior art systems. Also achieved is power usage advantage not available in prior art systems

There is no justification in Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., or any other prior art reference independent of applicants disclosure which suggests that these references be combined much less be combined in a manner proposed.

In addition, the proposed combination would not be physically possible or operative.

Even if the combinations of Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., be combined in a manner proposed, the proposed combination would not show all of the novel features of the present invention.

The new design concepts discovered by the applicant produce the new and unexpected results for the computer system; hence the concepts and resulting embodiments of the present invention are unobvious and patentable over these prior art references.

There is no suggestion in the three references, either in the form of some teaching, suggestion, incentive or inference in the applied references, or in the form of generally available knowledge that one having ordinary skill in the prior art would have been led to combine the relevant teachings of the applied reference in the proposed manner to arrive at the claimed invention and resolve the issue of unobviousness.

The applicant therefore submits that combining Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., is not legally justified and is therefore improper. Therefore the applicant submits the rejection on these references is also improper and should be with drawn.

Regarding claim 9, a logic for preventing simultaneous writing of the same location by more than one device is well known in the art ... in a system made obvious by the combination of Mekhiel (US6,587,920), Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999).

The applicant submits that claim 9 is a dependent claim. The dependent claim 9 incorporates all the subject matter of claim 1 and add additional subject matter which makes it fortiori and independently patentable over the cited references. The claim 9 is defined as having the basic elements of claim 1 which uses the novel concept of transition buffer to achieve deterministic operation without a cache memory and attendant cache memory miss problems. The prior art teachings proposed by the combination of Mekhiel (US6,587,920), Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999). does not use deterministic operation of transition buffer to obtain the data consistency and cache coherency function and also provide logic to prevent simultaneous writing of the same location by more than one device to prevent harmful contention as described in claims 1 and 9.

The prior art logic is based on probabilistic assumption that available data present in the cache memory and main memory are in agreement and the data available is the fresh data that can be used. This is not always true. Although the object of the logic to prevent simultaneous writing to one location by more than one device is intended to prevent contention, it happens that contention can occur in more than one area and even if contention is prevented, data integrity problems due to cache related problems of coherency and consistency in probabilistic systems is not avoided. While the CPU is trying to update data in main memory, the cache can have new data. And other device can assume that the data being updated is the fresh data. This causes data integrity and consistency problems. The CPU or the other device will not know in advance if the data in the two locations are consistent. The objective of the logic to prevent simultaneous writing to one location by more than one device is to prevent contention in the hope that no data integrity problem has occurred. However, in prior art systems according to Mekhiel (US6,587,920), Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999), there is no deterministic conclusion that the logic to prevent contention has also prevented data integrity problems due to cache coherency or consistency.

In the present invention, the data is always fresh. This is achieved because there are no two copies of data, as found in prior art systems one in cache memory and other in the main memory. The object of the logic to prevent simultaneous writing to one location by more than one device is to prevent other device from wasting time to read or write to this location while other device is using the data. There is no question as to the freshness or integrity of the data being processed.

The new and unexpected result produced here is that a device is prevented from spending time on an operation that will not succeed. This will avoid undesirable execution delay, which further adds to the higher performance of the architecture.

There is no justification in Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., or any other prior art reference independent of applicants disclosure which suggests that these references be combined much less be combined in a manner proposed.

In addition, the proposed combination would not be physically possible or operative.

Even if the combinations of Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., be combined in a manner proposed, the proposed combination would not show all of the novel features of the present invention.

The new design concepts discovered by the applicant produce the new and unexpected results for the computer system; hence the concepts and resulting embodiments of the present invention are unobvious and patentable over these prior art references.

There is no suggestion in the three references, either in the form of some teaching, suggestion, incentive or inference in the applied references, or in the form of generally available knowledge that one having ordinary skill in the prior art would have been led to combine the relevant teachings of the applied reference in the proposed manner to arrive at the claimed invention and resolve the issue of unobviousness.

The applicant therefore submits that combining Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., is not legally justified and is therefore improper. Therefore the applicant submits the rejection on these references is also improper and should be withdrawn.

Regarding claim 10, pre-decoding of instruction in CPU is well known in the art in order to enable better scheduling and prediction of program execution, and for this reason... by

combination of Mekhiel (US6,587,920), Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999).

The applicant submits that claim 10 is a dependent claim. The dependent claim 10 incorporates all the subject matter of claim 1 and add additional subject matter which makes it fortiori and independently patentable over the cited references. The claim 10 is defined as having the basic elements of claim 1 which uses the novel concept of transition buffer to achieve deterministic operation without a cache memory and attendant cache memory miss problems. The prior art teachings proposed by Mekhiel, Goodnow and Sen does not use deterministic operation of transition buffer to obtain the advance information available at compile time to decode instruction without pipeline stalls or cache miss along with data consistency and cache coherency function as described in claims 1 and 10.

The prior art logic is based on probabilistic assumption that available data present in the cache memory and main memory are in agreement and the data available is the fresh data that can be used. This is not always true.

The architecture of the present invention uses the novel concept of transition buffer to achieve deterministic operation without a cache memory and attendant cache memory miss problems. In the present invention due to deterministic operation of the computer system, it is possible to predict at compile time, if pre-decoded instruction will complete the pipeline without stalls. If the pre-decoded instruction is scheduled to complete the pipeline and produce the results, the pre-decoded instruction is allowed to proceed. This is an important feature that will allow prediction of performance problems in advance and avoid them or reschedule to avoid program stalls. The advance scheduling can be used with advantage to achieve higher performance and low power usage in complex instructions that has multiple dependencies. The problem of dependencies is handled very efficiently. There is no solution for pipeline stall that are related to dependencies in prior art systems. In view of high performance requirements of the present architecture over prior art, the restart problems cannot be allowed to degrade the performance. The deterministic operation of the computer system can be used to predict scheduling

problems at compile time. This will prevent unnecessary start of the instruction decode that will not be useful. The prior art systems using pipelined architectures are limited by the probabilistic behavior of the data, which can cause instruction stalls due to many events. Again, in the present invention this can be prevented with low power consumption and at a lower cost.

The deterministic execution of branch and jump instructions is important for real time execution of programs. This is done without execution delays or real-time data error in the present invention.

The architecture of the present invention goes beyond simple implementation of pre-decode for better scheduling. The use of transition buffer along with deterministic information available at compile time allows real-time operation even with complex instructions. This feature is not found in prior art systems. Also achieved is power usage advantage not available in prior art systems

The new and unexpected result produced here is that a device is prevented from spending time on an operation that will not succeed. This will avoid undesirable execution delay, which further adds to the higher performance of the architecture.

There is no justification in Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., or any other prior art reference independent of applicants disclosure which suggests that these references be combined much less be combined in a manner proposed.

In addition, the proposed combination would not be physically possible or operative.

Even if the combinations of Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., be combined in a manner proposed, the proposed combination would not show all of the novel features of the present invention.

The new design concepts discovered by the applicant produce the new and unexpected results for the computer system; hence the concepts and resulting embodiments of the present invention are unobvious and patentable over these prior art references.

There is no suggestion in the three references, either in the form of some teaching, suggestion, incentive or inference in the applied references, or in the form of generally available knowledge that one having ordinary skill in the prior art would have been led to combine the relevant teachings of the applied reference in the proposed manner to arrive at the claimed invention and resolve the issue of unobviousness.

The applicant therefore submits that combining Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., is not legally justified and is therefore improper. Therefore the applicant submits the rejection on these references is also improper and should be withdrawn.

Regarding claim 14, it is well known in the art to transfer program and data segments from peripheral devices including hard drives .....to use DMA to transfer data and instructions into the buffer and main memory in the system made obvious by the combination of Mekhiel (US6,587,920), Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999).

The applicant submits that claim 14 is a dependent claim. The dependent claim14 incorporates all the subject matter of claim 1 and add additional subject matter which makes it fortiori and independently patentable over the cited references. The claim 14 is defined as having the basic elements of claim 1 which uses the novel concept of transition buffer to achieve deterministic operation without a cache memory and attendant cache memory miss problems. The prior art teachings proposed by the combination of Mekhiel (US6,587,920), Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999).

by does not use deterministic operation of transition buffer to implement program transferred by DMA to allow CPU to continue executing instructions in parallel with the transfer of data and also obtain the data consistency and cache coherency function as described in claims 1 and 14.

The prior art logic is based on probabilistic assumption that available data present in the cache memory and main memory are in agreement and the data available is the fresh data that can be used. This is not always true.

The cache memory based prior art computer systems execute instructions from the cache memory. In a normal mode, data is first transferred to the main memory and then from the main memory data is transferred to the cache. The CPU is not setup to execute instructions from the main memory. If cache memory is receiving data from the DMA, the CPU is not able to access cache.

It is also important that the required data is available in the main memory for cache to use in the event a cache miss occurs.

In the architecture according to the present invention, instruction can be executed from both the main memory as well as from the transition buffer. Because of the inherent parallelism preferred in the main memory and transition buffer, it is possible to transfer data using DMA to the parts of main memory as well as transition buffer without suspending operation and instruction execution by CPU.

Simultaneous transfer of data from DMA to cache as well as the main memory in prior art is not known. This is because of the assumption that the required data is available from the cache during all the time and if the data is not available a miss occurs and main memory will be accessed. This is the type of probabilistic assumption that the available data present in the cache memory is the actual data needed that leads to system stalls and problems related to cache memory miss well known in prior art.

In addition it is also possible to predict problems with required data transfer mechanism at the compile time. At the compile time DMA operation can be verified for proper requirements. If there are any conflicts in the data transferred by DMA with the CPU, it is possible determine this conflict in advance.

The architecture of the present invention goes beyond simple implementation of DMA transfer to the memory for execution. The use of transition buffer along with deterministic information available at compile time allows real-time operation even with DMA data transfer. This feature is not found in prior art systems. Also achieved is power usage advantage not available in prior art systems
The new and unexpected result produced here is that a device is prevented from spending time on an operation that will not succeed. This will avoid undesirable execution delay, which further adds to the higher performance of the architecture.

There is no justification in Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., or any other prior art reference independent of applicants disclosure which suggests that these references be combined much less be combined in a manner proposed.

In addition, the proposed combination would not be physically possible or operative.

Even if the combinations of Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., be combined in a manner proposed, the proposed combination would not show all of the novel features of the present invention.

The new design concepts discovered by the applicant produce the new and unexpected results for the computer system; hence the concepts and resulting embodiments of the present invention are unobvious and patentable over these prior art references.

There is no suggestion in the three references, either in the form of some teaching, suggestion, incentive or inference in the applied references, or in the form of generally available knowledge that one having ordinary skill in the prior art would have been led to combine the relevant teachings of the applied reference in the proposed manner to arrive at the claimed invention and resolve the issue of unobviousness.

The applicant therefore submits that combining Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., is not legally justified and is therefore improper. Therefore the applicant submits the rejection on these references is also improper and should be withdrawn.

Regarding claim 16, it well known in the art... Mekhiel (US6,587,920), Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999).

The applicant submits that claim 16 is a dependent claim. The dependent claim 16 incorporates all the subject matter of claim 1 and add additional subject matter which makes it fortiori and independently patentable over the cited references. The claim 16 is defined as having the basic elements of claim 1 which uses the novel concept of transition buffer to achieve deterministic operation without a cache memory and attendant cache memory miss problems. The prior art teachings proposed by does not use deterministic operation of transition buffer and compile time information to optimize power usage by constantly keeping off components when not required in a deterministic manner. At the same time the components are turned on only when required at the most optimum time to obtain the maximum consistency in power usage by reducing heat dissipation thereby optimizing thermal management as described in claims 1 and 16.

The prior art logic is based on probabilistic assumption that available data present in the cache memory and main memory are in agreement and the data available is the fresh data that can be used. This is not always true.

The prior art system uses power management as best possible effort to turn off components when not required. In a probabilistic system it is impossible to predict when a component or a subsystem will be required in a given sequence of program events. This leads to periods when subsystems not in use will still be provided with power.
Power management is not inherently programmed to act with the execution of the program.
The present invention implements power management by estimating and predicting power usage and heat dissipation at the compile time by use of deterministic operation of transition buffer and compile time information to optimize power usage by constantly keeping off components when not required in a deterministic manner. At the same time the components are turned on only when required at the most optimum time to obtain the maximum consistency in power usage by reducing heat dissipation thereby optimizing thermal management.
In fact it is possible to predict power usage and thermal behavior at the compile time and query behavior responses and make necessary adjustment at the compile time. Best possible power usage profile can be developed before program execution.
The new and unexpected result produced here is that a device is prevented from spending time on an operation that will not succeed. This will avoid undesirable execution delay, which further adds to the higher performance of the architecture.

There is no justification in Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., or any other prior art reference independent of applicants disclosure which suggests that these references be combined much less be combined in a manner proposed.

In addition, the proposed combination would not be physically possible or operative.

Even if the combinations of Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., be combined in a manner proposed, the proposed combination would not show all of the novel features of the present invention.

The new design concepts discovered by the applicant produce the new and unexpected results for the computer system; hence the concepts and resulting embodiments of the present invention are unobvious and patentable over these prior art references.

There is no suggestion in the three references, either in the form of some teaching, suggestion, incentive or inference in the applied references, or in the form of generally available knowledge that one having ordinary skill in the prior art would have been led to combine the relevant teachings of the applied reference in the proposed manner to arrive at the claimed invention and resolve the issue of unobviousness.

The applicant therefore submits that combining Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., is not legally justified and is therefore improper. Therefore the applicant submits the rejection on these references is also improper and should be withdrawn.

Claim 19 is rejected using the same rationale as for rejection of claim 1, and further noting that it is well known in the art to configure a plurality of CPU's with a shared memory system in order to improve processing performance while allowing sharing of data among processors.

The applicant would like to respectfully request that the rejection of claim 19 should be withdrawn on the same basis as requested for claim 1.

The applicant would like to point out that prior art multiprocessing architectures still uses cache based approach. As indicated in the specification, the prior art processing systems

whether single processor or multiprocessor systems, requires to have more than one copy of data. The correct status of every copy of data requires to be updated every time data in cache memory is updated. The data present in main memory is not always fresh. This creates data consistency and coherency problems as it relates to cache memory. Ample literature is available in prior art, which describes the problems this creates in cache memory based systems. The problems relating to data consistency and coherency is magnified many times due to large number of processor communicating with each other for multiprocessor operation. Most of the interprocessor communication is relating to status and change of data and needed communication requirements for each processor. If the data available on the multiprocessing architecture is not fresh all the time, the processor will have to wait till the data is updated. In addition, it introduces a large number of interprocessor communication events via shared memory or directly with each other, which further introduces execution delays and data integrity problems.

The use of transition buffer for storage of data assures that data available is always fresh. This produces an advantage over prior art shared memory systems. The new and unexpected results produced by the use of transition buffer in multiprocessor environment makes the present invention unobvious and hence patentable.

The shared memory multiprocessor systems and distributed memory multiprocessor systems each have their own applications and advantages.

Shared memory multiprocessor system according to the present invention is described in the specification as one of many possible embodiments. The advantages of the present invention is equally applicable to distributed memory systems, where each processor has its own memory and each processor communicates with other processor using an interconnect structure. The prior art distributed memory system has more severe problems in that architecture due to a large number of interprocessor communication events needed just to keep up with the integrity and consistency status of data each processor uses.

The use of transition buffer will greatly reduce this interprocessor communication and hence improve the overall performance of the system. It will also guarantee the integrity of data handled by each processor.
This again is an orders of magnitude improvement over the prior art multiprocessor distributed computing systems.

There is no justification in Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., or any other prior art reference independent of applicants disclosure which suggests that these references be combined much less be combined in a manner proposed.

In addition, the proposed combination would not be physically possible or operative.

Even if the combinations of Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., be combined in a manner proposed, the proposed combination would not show all of the novel features of the present invention.

The new design concepts discovered by the applicant produce the new and unexpected results for the computer system; hence the concepts and resulting embodiments of the present invention are unobvious and patentable over these prior art references.

There is no suggestion in the three references, either in the form of some teaching, suggestion, incentive or inference in the applied references, or in the form of generally available knowledge that one having ordinary skill in the prior art would have been led to combine the relevant teachings of the applied reference in the proposed manner to arrive at the claimed invention and resolve the issue of unobviousness.

The applicant therefore submits that combining Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., is not legally justified and is therefore

improper. Therefore the applicant submits the rejection on these references is also improper and should be withdrawn.

Claims 18 and 20 are rejected using the same rationale as for the rejection of claim 9.

The applicant would like to respectfully request the withdrawal of rejection of claims 18 and 20 on the same basis as requested for claim 9.

The applicant submits that claim 18 is a dependent claim. The dependent claim 18 incorporates all the subject matter of claim 17 and add additional subject matter which makes it fortiori and independently patentable over the cited references. The claim 18 is defined as having the basic elements of claim 17, which uses the novel concept of transition buffer to achieve deterministic operation without a cache memory and attendant cache memory miss problems. The prior art teachings proposed by the combination of Mekhiel (US6,587,920), Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., does not use deterministic operation of transition buffer to obtain the data consistency and cache coherency function so that logic and storage area in the CPU is used to obtain data that can be available to central processing unit (CPU) at any time and at the same time the data can be related to the main data memory and prevent simultaneous writing of same location by more than one device as described in claims 1,9, 17 and 18.

The dependent claim 18 claims a logic and storage area in the CPU. This includes a logic and storage area to store data that can be available to central processing unit (CPU) at any time and at the same time the data can be related to the main data memory. The CPU is also provided with logic to prevent simultaneous writing of same location by more than one device.

As indicated earlier in the response to claim 9, in the present invention, the data is always fresh. This is achieved because there are no multiple copies of data, as found in prior art

systems, one in cache memory and other in the main memory. The object of the logic to prevent simultaneous writing to one location by more than one device is to prevent other device from wasting time to read or write to this location while other device is using the data. There is no question as to the freshness or integrity of the data being processed.

However, in prior art systems according to Mekhiel (US6,587,920), Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999), there is no deterministic conclusion that the logic to prevent contention has also prevented data integrity problems due to cache coherency or consistency.

The new and unexpected result produced here is that a device is prevented from spending time on an operation that will not succeed. This will avoid undesirable execution delay, which further adds to the higher performance of the architecture.
There is no justification in Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., or any other prior art reference independent of applicants disclosure which suggests that these references be combined much less be combined in a manner proposed.

In addition, the proposed combination would not be physically possible or operative.

Even if the combinations of Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., be combined in a manner proposed, the proposed combination would not show all of the novel features of the present invention.

The new design concepts discovered by the applicant produce the new and unexpected results for the computer system; hence the concepts and resulting embodiments of the present invention are unobvious and patentable over these prior art references.

There is no suggestion in the three references, either in the form of some teaching, suggestion, incentive or inference in the applied references, or in the form of generally available knowledge that one having ordinary skill in the prior art would have been led to combine the relevant teachings of the applied reference in the proposed manner to arrive at the claimed invention and resolve the issue of unobviousness.

The applicant therefore submits that combining Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., is not legally justified and is therefore improper. Therefore the applicant submits the rejection on these references is also improper and should be withdrawn.

The applicant submits that claim 20 is a dependent claims. The dependent claim20 incorporates all the subject matter of claim 19 and add additional subject matter which makes it fortiori and independently patentable over the cited references. The claim 20 is defined as having the basic elements of claim 19, which uses the novel concept of transition buffer to achieve deterministic operation without a cache memory and attendant cache memory miss problems. The prior art teachings proposed by the combination of Mekhiel (US6,587,920), Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., does not use deterministic operation of transition buffer to obtain the data consistency and cache coherency function and also provide logic to prevent simultaneous writing of the same location by more than one device to prevent harmful contention as described in claims 1, 9 and 19.

The dependent claim 20 claims that the CPU is provided with logic to prevent simultaneous writing of same location by more than one device.

The applicant would like to point out that prior art multiprocessing architectures still uses cache based approach. As indicated in the specification, the prior art processing systems whether single processor or multiprocessor systems, requires to have more than one copy

of data. The data status of every copy of data requires to be updated every time data in cache memory is updated. The data present in main memory is not always fresh. This creates data consistency and coherency problems as it relates to cache memory. Ample literature is available in prior art, which describes the problems this creates in cache memory based systems. The problems relating to data consistency and coherency is magnified many time due to large number of processor communicating with each other for proper operation. Most of the interprocessor communication is relating to data requirements for each processor. If the data available on the multiprocessing architecture is not fresh all the time, the processor will have to wait till the data is updated. In addition, it introduces a large number of interprocessor communication events via shared memory or directly with each other, which further introduces execution delays and data integrity problems.

The use of transition buffer for storage of data assures that data available is always fresh. This produces an advantage over prior art shared memory systems. The new and unexpected results produced by the use of transition buffer makes the present invention unobvious and hence patentable.

The shared memory multiprocessor systems and distributed memory multiprocessor systems each have their own applications and advantages.

Shared memory multiprocessor system according to the present invention is described in the specification as one of many possible embodiments. The advantages of the present invention is equally applicable to distributed memory systems, where each processor has its own memory and each processor communicates with other processor using an interconnect structure. The prior art distributed memory system has more severe problems in that architecture due to a large number of interprocessor communication events needed just to keep up with the integrity and consistency status of data each processor uses.

The use of transition buffer will greatly reduce this interprocessor communication and hence improve the overall performance of the system. It will also guarantee the integrity of data handled by each processor.
This again is an orders of magnitude improvement over the prior art multiprocessor distributed computing systems.

As indicated earlier in the response to claim 9, in the present invention, the data is always fresh. This is achieved because there are no multiple copies of data, as found in prior art systems, one in cache memory and other in the main memory. The object of the logic to prevent simultaneous writing to one location by more than one device is to prevent other device from wasting time to read or write to this location while other device is using the data. There is no question as to the freshness or integrity of the data being processed.
In multiprocessor systems whether a shared memory system or a distributed systems, the logic to prevent simultaneous writing of more than one device to single location saves important execution time because a large number of processor can request the same data simultaneously and if they are not properly scheduled, execution time is wasted.
However, in prior art systems according to Mekhiel (US6,587,920), Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999), there is no deterministic conclusion that the logic to prevent contention has also prevented data integrity problems due to cache coherency or consistency.

The new and unexpected result produced here is that a device is prevented from spending time on an operation that will not succeed. This will avoid undesirable execution delay, which further adds to the higher performance of the architecture.

There is no justification in Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., or any other prior art reference independent of applicants disclosure which suggests that these references be combined much less be combined in a manner proposed.

In addition, the proposed combination would not be physically possible or operative.

Even if the combinations of Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., be combined in a manner proposed, the proposed combination would not show all of the novel features of the present invention.

The new design concepts discovered by the applicant produce the new and unexpected results for the computer system; hence the concepts and resulting embodiments of the present invention are unobvious and patentable over these prior art references.

There is no suggestion in the three references, either in the form of some teaching, suggestion, incentive or inference in the applied references, or in the form of generally available knowledge that one having ordinary skill in the prior art would have been led to combine the relevant teachings of the applied reference in the proposed manner to arrive at the claimed invention and resolve the issue of unobviousness.

The applicant therefore submits that combining Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999)., is not legally justified and is therefore improper. Therefore the applicant submits the rejection on these references is also improper and should be withdrawn.

Claim 7 is rejected under 35 USC 103(a) as being unpatentable over Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999). As applied to claim 1 as above, and further in view of Young(cliff Young et al, "near optimal intraprocedural branch alignment," ACM 1997) and examiners official notice.

The applicant would like to respectfully request the withdrawal of rejection of claims 7 on the same basis as requested for claim 1.

The applicant submits that claim 7 is a dependent claim. The dependent claim7 incorporates all the subject matter of claim 1 and add additional subject matter which makes it fortiori and independently patentable over the cited references. The claim 7 is defined as having the basic elements of claim 1 which uses the novel concept of transition buffer to achieve deterministic operation without a cache memory and attendant cache memory miss problems. The prior art teachings of Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999). As applied to claim 1 as above, and further in view of Young(cliff Young et al, "near optimal intraprocedural branch alignment," ACM 1997) does not teach logic in main execution memory means wherein sequential starting locations of multiple branch instructions are stored for implementing program branch instructions as they occur in program execution in required order and determined at the compile time, so that the starting locations for a new branch will be available to said central processing unit from this pipelined storage architecture by the use of interleaved memory to obtain the deterministic and real-time functions as described in claim 1 and 7.

Interleaved memory has been used in prior art to hide latency in memory. The function of interleaved memory has been to provide access and transfer to only consecutive locations of memory, generally in a burst mode.
Interleaved memory cannot execute a jump or branch instruction without a delay in execution. If the CPU requires an access to a non-sequential location on an interleaved memory to implement instruction such as a branch or jump, an execution delay equal to access time of the next group of memory locations will be introduced. This will make the interleaved memory as taught in prior art and as described in the official notice incapable of handling branch instructions. The implementation of deterministic multiway branch instructions without execution delay are one of the strengths of the present invention. The prior art architecture of the interleaved memory to hide latency will not be able to implement the multiway branch without cache memory miss if the required location is not found on the cache memory. If the interleaved memory can allow CPU to execute

jump and branch instructions, there will be no need for cache memory or any other architecture to fill the gap between the CPU cycle time and the memory access time as has been used in prior art.

The examiner further writes, "Young teaches a compile time reordering of program blocks where basic blocks .....a pipeline fed by fetching the code sequence."
The applicant would like to respectfully point out that as discussed earlier on analysis of Young paper earlier, the object of this paper is to conduct best possible analysis of reordering of program blocks in a probabilistic system. The result is still a probabilistic estimation of CTI behavior.
Page 184 line 8 it refers to misfetch penalties and preloading a cache based system with a best possible estimation of future branches.
The Young paper suggests a solution to a problem the present invention does not have. As indicated on line 1 of the abstract the branch alignment reorders the basic blocks of program to minimize pipeline penalties due to control transfer instructions. Misfetch and cache miss occurs on probabilistic systems where advance information in time is not available as to the address of the next instruction. Even with the best possible reordering of the program blocks, misfetch and cache misses cannot be prevented. This is the inherent nature of all the probabilistic systems. There is an unobvious difference between reducing misfetches and cache misses and eliminating them. The deterministic operation and compile time information allows the CPU to eliminate misfetch altogether with the new concept. The availability of required data on the transition buffer during all the fetch conditions will not require any compile time reordering as suggested by Young.

In claim 7 the multiway branch pipeline is implemented in the main execution memory. As a normal or dedicated CPU function, the multiway branch pipeline can be implemented in the CPU core with the advance availability of information at the compile time. However, due to the ability of the interleaved parallel banks, the new concept will allow access to any location on the main execution memory, this function is implemented in main execution memory according to claim 7.

There is no justification in Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999). As applied to claim 1 as above, and further in view of Young(cliff Young et al, "near optimal intraprocedural branch alignment," ACM 1997) and examiners official notice., or any other prior art reference independent of applicants disclosure which suggests that these references be combined much less be combined in a manner proposed.

In addition, the proposed combination would not be physically possible or operative.

Even if the combinations of Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999). As applied to claim 1 as above, and further in view of Young(cliff Young et al, "near optimal intraprocedural branch alignment," ACM 1997) and examiners official notice be combined in a manner proposed, the proposed combination would not show all of the novel features of the present invention.

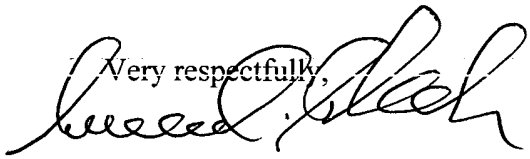
The new design concepts discovered by the applicant produce the new and unexpected results for the computer system; hence the concepts and resulting embodiments of the present invention are unobvious and patentable over these prior art references.

There is no suggestion in the three references, either in the form of some teaching, suggestion, incentive or inference in the applied references, or in the form of generally available knowledge that one having ordinary skill in the prior art would have been led to combine the relevant teachings of the applied reference in the proposed manner to arrive at the claimed invention and resolve the issue of unobviousness.

The applicant therefore submits that combining Mekhiel (US6,587,920) in view of Goodnow (US6,587,920) and Sen (Subhabrata Sen et al, "Proxy Prefix Caching for Multimedia Streams", IEEE, March 1999). As applied to claim 1 as above, and further in view of Young(cliff Young et al, "near optimal intraprocedural branch alignment," ACM

1997) and examiners official notice, is not legally justified and is therefore improper. Therefore the applicant submits the rejection on these references is also improper and should be withdrawn.

Very respectfully,



(EMANUEL E SHAH)

11606B Argonne forest trail
Austin Texas 78759
Telephone: 512.331.1383